This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Original) An apparatus, comprising:

a message dispatcher that routes and dispatches messages, wherein each message is routed based on an arbitrary portion of the message's contents; and

an interface through which application programs communicate with the message dispatcher to define the arbitrary portion of the message's contents.

2. (Original) The apparatus of claim 1, wherein the message dispatcher comprises a transport independent message dispatcher, and the message dispatcher communicates using a transport independent protocol.

3. (Original) The apparatus of claim 1, wherein the message dispatcher routes a first network message based on a first attribute of said first network message, and routes a second network message based on a second attribute, different from said first attribute, of said second network message.

4. (Original) The apparatus of claim 1 wherein the message dispatcher routes a first network message, addressed to a recipient from a first sender, to a first server, and

wherein the message dispatcher routes a second network message, addressed to the recipient from a second sender, to a second server.

5. (Original) The apparatus of claim 1, wherein the message dispatcher routes messages using a virtual network protocol above a transport layer protocol.

6. (Currently Amended) The apparatus of claim ~~1~~5, further comprising a transport adapter to convert messages between the transport layer protocol and the virtual network protocol.

7. (Original) The apparatus of claim 1, wherein the arbitrary portion of the message's contents comprises an application level header.

8. (Currently Amended) A data processing apparatus, comprising:

a message dispatcher module;

a transport adapter for interfacing the message dispatcher to a transport protocol;

an interface through which application programs communicate with the message dispatcher module;

stored rules instructing the message dispatcher to route a first network message based on a first arbitrary attribute of said first network message, and route a second network message based on a second arbitrary attribute, different from said first arbitrary attribute, of said second network message, wherein the first and second arbitrary attributes are selected from a set of headers and data contained in each network message.

9. (Currently Amended) The data processing apparatus of claim 8, wherein the first arbitrary attribute comprises an application created header.

10. (Currently Amended) The data processing apparatus of claim 8, wherein each ~~message~~ stored rule is stored in a message handler.

11. (Currently Amended) The data processing apparatus of claim 10, comprising a first message handler that, upon the occurrence of a predetermined condition, sends an alteration message to alter ~~alters~~ a second message handler.

12. (Currently Amended) The data processing apparatus of claim 10, comprising a first message handler that, upon the occurrence of a predetermined condition, alters the first network message.

13. (Original) The data processing apparatus of claim 11, wherein the predetermined condition comprises a nonoccurrence of an event.

14. (Currently Amended) The data processing apparatus of claim 13, wherein the message dispatcher module comprises computer executable instructions that, when executed, cause the data processing apparatus to perform the steps of:

(i) polling a second apparatus in first predetermined intervals; and

(ii) receiving poll responses from the second apparatus;

and wherein the predetermined condition comprises the nonoccurrence of step (ii) for a predetermined amount of time.

15. (Original) The data processing apparatus of claim 14, wherein when the predetermined condition is met, the message dispatcher alters the second message handler to redirect messages, that were originally addressed to the second apparatus, to a third apparatus.

16. (Original) The data processing apparatus of claim 15, wherein the computer executable instructions further cause the data processing apparatus to perform the step of sending routing information to a second message dispatcher, indicating the change of routing information corresponding to the second and third apparatus.

17. (Currently Amended) A method for routing network messages, comprising the steps of:

(i) routing a first network message based on a first attribute of the first network message;

(ii) routing a second network message based on a second attribute, different from said first attribute, of said second network message;

wherein the first and second attributes ~~may be any field~~ are arbitrarily selected from a set of headers and data of each network message.

18. (Original) The method of claim 17, further comprising the steps of:

(iii) receiving instructions comprising a message field and a field condition;

(iv) modifying a message handler based on the received instructions.

19. (Original) The method of claim 18, wherein, in step (iii), the instructions are received from a network application program.

20. (Original) The method of claim 18, wherein, in step (iii), the instructions are based on user-input.

21. (Original) The method of claim 17, wherein, in steps (i) and (ii), each message is output to a transport adapter that converts the message from a virtual network protocol to a transport protocol.

22. (Original) The method of claim 17, wherein, in step (i), the first attribute comprises an application created header.

23. (Original) The method of claim 17, further comprising the step of storing routing instructions in message handlers, and

wherein steps (i) and (ii) are performed based on stored message handlers.

24. (Original) The method of claim 23, further comprising the step of altering a first message handler when a predetermined condition occurs.

25. (Original) The method of claim 23, further comprising the step of altering a network message when the message meets a predetermined condition stored in a message handler.

26. (Original) The method of claim 24, wherein the predetermined condition comprises a nonoccurrence of an event.

27. (Original) The method of claim 17, further comprising the steps of:

(iii) polling a first data processing device in predetermined intervals;

(iv) receiving poll responses from the first data processing device; and

(v) when step (iv) has not occurred for a predetermined amount of time, altering a message handler to direct messages originally addressed to the first data processing device, to a second data processing device.

28. (Original) The method of claim 27, further comprising the step of sending routing information to a message dispatcher, indicating the change of routing information corresponding to the first and second data processing devices.

29. (Currently Amended) A network router comprising computer executable instructions that, when executed by the router, perform steps of:

(i) storing routing information received from a network application, wherein the routing information comprises a message field, a field condition, and a routing instruction;

(ii) receiving a network message;

(iii) processing the network message by comparing the network message to the stored routing information;

(iv) when ~~the received message's~~ a message field of the received message meets the field condition, performing the routing instruction.

30. (Currently Amended) The network router of step 29, wherein, in step (iv), the routing instruction comprises altering the network message.

31. (Currently Amended) The network router of step 29, wherein, in step (iv), the routing instruction comprises routing the message based on an application level header.

32. (Currently Amended) A computer network, comprising:

a plurality of computers, each comprising:

at least one transport adapter that converts messages between a transport layer protocol and a network protocol; and

a message dispatcher that routes and dispatches messages based on an arbitrary portion of the message's contents, and wherein the message dispatcher in each computer routes messages in ~~the~~ a virtual network protocol over the transport layer protocol using the at least one transport adapter.

33. (Original) The computer network of claim 32, wherein a first message dispatcher in a first computer is configurable for use with a new transport protocol by adding a new transport adapter that converts messages between the new transport layer protocol and the network protocol, without requiring a network application to be reconfigured for use with the new transport protocol.

34. (Currently Amended) A virtual network, comprising at least one virtualized component inserted between layer 7 and layer 6 of an OSI protocol stack, wherein said virtualized component provides a virtual network service.

35. (Currently Amended) The virtual network of claim 34, wherein the at least one virtualized component comprises a virtual network message dispatcher to route messages according to virtual names and locations.

36. (Currently Amended) The virtual network of claim 34, wherein the at least one virtualized component comprises a synchronization module to ensure that distributed data within the virtual network remains synchronized.

37. (Currently Amended) The virtual network of claim 34, wherein the at least one virtualized component comprises an eventing module to create new routing and dispatch rules based on an occurrence or non-occurrence of one or more events.

38. (Currently Amended) The virtual network of claim 34, wherein the at least one virtualized component comprises a names module to provide name resolution services based on any substring of a virtual name.

39. (Currently Amended) The virtual network of claim 34, wherein the at least one virtualized component comprises a groups module to manage name-mapping tables.

40. (Currently Amended) The virtual network of claim 34, wherein the at least one virtualized component comprises an addressing module to perform naming and routing services for fixed-length address names.

41. (Currently Amended) The virtual network of claim 34, wherein the at least one virtualized component comprises a security module to ensure that message contents are secure and authentic.

42. (Currently Amended) The virtual network of claim 34, wherein the at least one virtualized component comprises an administrative module to monitor network performance and usage.

43. (New) The method of claim 17, further comprising the step of:

(iii) adding a new message handler to route messages based on a newly created type of message header.

44. (New) The method of claim 17, wherein either of the first or second attributes correspond to a geographic location of the sender of the message.

45. (New) The method of claim 17, wherein either of the first or second attributes correspond to a class of service of the sender of the message.

46. (New) A computer network architecture comprising a plurality of data processing devices interconnected via a computer network, each data processing device comprising:

a virtual message dispatcher that routes messages to intended destinations and dispatches messages to appropriate applications at their intended destination, wherein each message is handled based on an arbitrary portion of the message's contents; and

an interface through which OSI layer 7 application programs communicate with the message dispatcher to define the arbitrary portion of the message's contents by which each message is handled,

wherein the virtual message dispatcher comprises a transport adapter for converting messages between a virtual network protocol used by network applications and a transport protocol used by the computer network, and

wherein the virtual message dispatcher is configurable for use with a second transport protocol by adding a second transport adapter that converts messages between the second transport protocol and the virtual network protocol, without requiring any network applications to be reconfigured for use with the second transport protocol.